WHAT IS CLAIMED IS:

1. A processor comprising:

a first storage circuit comprising a free list, the free list including a set of physical register names and a corresponding set of indications each associated with one of the physical register names, wherein each indication is indicative, in a first state, that: (i) a physical register associated with a corresponding physical register name was assigned to store a register result and to store a flag result of a first instruction, and (ii) a subsequent instruction overwrites a logical register corresponding to a register result of the first instruction but not the flag result, and wherein each indication is indicative, in a second state, that at least one of (i) or (ii) is not detected, and wherein the first storage circuit is configured to output at least a first physical register name and a first indication corresponding to the first physical register name;

a second storage circuit configured to store one or more physical register names separate from the free list, wherein the second storage circuit is configured to output a second physical register name; and

a selection circuit coupled to receive the first physical register name from the first storage circuit and the second physical register name from the second storage circuit and further configured to receive the first indication, wherein the selection circuit selects the first physical register name as a selected physical register name responsive to the first indication being in the first state and selects the second physical register name as the selected physical register name responsive to the first indication being in the second state;

wherein the selected physical register name is provided to a mapper circuit for assignment to a logical register.

2. The processor as recited in claim 1, wherein the mapper circuit includes most recent writer circuitry configured to store a name of a most recently written logical register name that was a destination of a most recent instruction which also updated the flags, wherein the most recent writer circuitry is coupled to receive one or more destination logical register names corresponding to instructions being presented to the mapper circuitry, and wherein the most recent writer circuitry is configured to determine if one of the destination logical register names is equivalent to a most recently written logical register name.

3. The processor as recited in claim 2, wherein the mapper circuit includes a current map which indicates a correspondence between physical register names and logical register names, wherein the most recent writer circuitry is further configured to provide an indication to the current map if the destination logical register name is equivalent to the most recently written logical register name and a corresponding instruction does not update flag results.

4. The processor as recited in claim 3, wherein the current map is coupled to receive physical register names from the selection circuit and is configured to return physical register names to the free list for physical registers that have been removed from the current map by subsequent assignments of physical register names.

5. The processor as recited in claim 1, wherein the first physical register name is stored in the second storage circuit if the first indication is in the second state.

6. The processor as recited in claim 1, wherein the free list is a circular buffer, wherein an insert pointer points to a first address in the free list.

7. The processor as recited in claim 6, wherein physical register names are inserted into the free list beginning at the insert pointer.

8. The processor as recited in claim 6, wherein physical register names are read from the free list beginning at the insert pointer.

9. A method for operating a processor, the method comprising:

storing a set of physical register names in a free list;

storing a corresponding set of indications in the free list, each indication associated with one of the physical register names in the free list;

wherein each indication is indicative, in a first state, that, (i) a physical register associated with a corresponding physical register name was assigned to store a register result and to store a flag result of a first instruction, and (ii) a subsequent instruction overwrites a logical register corresponding to a register result but not the flag result, and wherein each indication is indicative, in a second state, that at least one of (i) or (ii) is not detected;

outputting at least a first physical register name and a first indication corresponding to the first physical register name from the free list;

storing one or more physical register names separate from the free list;

outputting a second physical register name of the one or more physical register names;

selecting the first physical register name as a selected physical register name responsive to the first indication being in the first state;

selecting the second physical register name as the selected physical register name responsive to the first indication being in the second state; and

providing the selected physical register name for assignment to a logical register.

10. The method as recited in claim 9 further comprising receiving one or more destination logical register names, storing a most recently written logical register

name that was a destination of a most recent instruction which also updated the flags, and determining if one of the one or more destination logical register names is equivalent to the most recently written logical register name.

11. The method as recited in claim 10, further comprising providing an indication if one of the one or more destination logical register names is equivalent to the most recently written logical register name and an instruction corresponding to the one of the one or more destination logical register names does not update flag results.

12. The method as recited in claim 11 further comprising indicating a correspondence between physical register names and logical register names.

13. The method as recited in claim 9 further comprising storing the first physical name separate from the free list if the first indication is in the second state.

14. The method as recited in claim 9 further comprising an insert pointer pointing to a first address in the free list.

15. The method as recited in claim 14 further comprising inserting physical register names into the free list at the first address.

16. The method as recited in claim 14 further comprising reading physical register names from the free list at the first address.